AN EDUCATIONAL SERVICES ARCHITECTURE TO SHARE E-LEARNING RESOURCES

Shanshan Yang, Mike Joy
Department of Computer Science, University of Warwick, Coventry, UK
Shanshan.Yang@warwick.ac.uk, M.S.Joy@warwick.ac.uk

Teemu H. Laine
School of Computing, University of Eastern Finland, Joensuu, Finland
teemu.laine@uef.fi

Keywords: E-learning, Service Oriented Architecture, Educational Services, Share Resources, Interoperability, Discoverability, Reusability.

Abstract: Today many quality e-learning applications have been developed. They are widely used to cover varied aspects of learning and teaching activities, such as developing learning materials, delivering learning activities, and performing assessment tasks. Different applications rely on different technologies, require different access characteristics, and are distributed on different servers across many organisations. However, people cannot fully benefit from these valuable resources as they have not been shared effectively and efficiently — there are many varieties of these resources, but these have not been properly described and linked. In this paper, we propose a service oriented approach as a potential solution to better sharing and reuse of e-learning resources, because it allows resources to be better discovered, accessed and linked.

1 INTRODUCTION

E-learning resources refer to e-learning applications and their content that support learning activities including delivering learning activities, developing learning materials and assessing learning performances. The typical content these applications handle include e-learning materials, assessment submissions, marks, students' records, and course information (Zhou *et al.*, 2009). In our research, sharing e-learning resources means redistribution, remix and reuse of currently available e-learning systems and their supported content.

The rapid development of the Internet has created opportunities to offer various e-learning resources that rely on different technologies (Su *et al.*, 2005). However, there are barriers to sharing and reusing these e-learning resources and include the following problems in current practices:

- e-learning resources are difficult to be discovered, as they are described poorly;
- e-learning resources are difficult to interoperate, as they rely on different implementation technologies; and
- e-learning resources are difficult to be reused, as they have not been shared well enough.

We propose a novel service oriented architecture as a potential solution for these issues, since it supports the principles of discoverability, interoperability and reusability. It allows resources to be outwardly described and linked, so that they can be better found, accessed and reused, without much work being required for reimplementation.

In e-learning, *interoperability* means that varied educational systems are connected so that requests and responses can be transmitted using standard messages, and passed around easily between those systems (Bean, 2010). *Discoverability* means that information about each application is described and stored in a service registry, so that potential users are able to search and compare available systems by querying the service registry. *Reusability* means that both students and teachers are able to reuse more

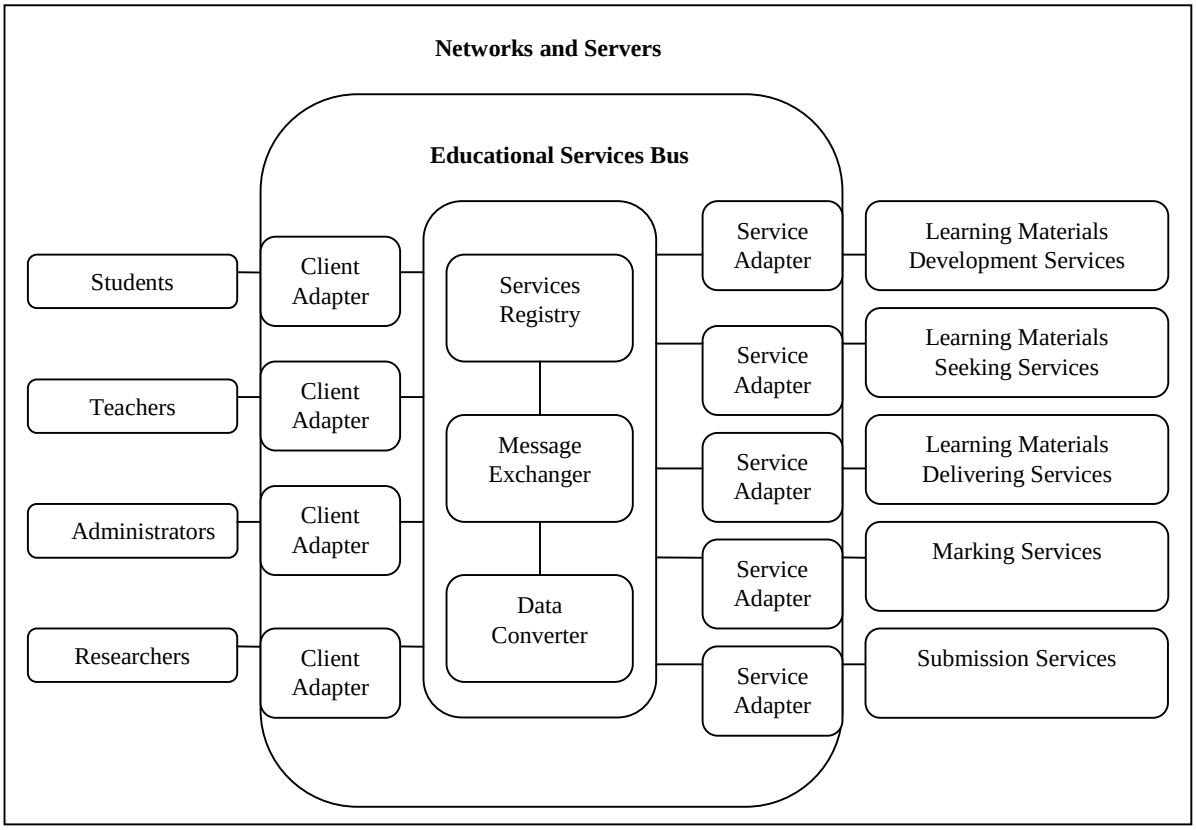

Figure 1: Educational Services Architecture

quality e-learning resources if they can be better shared (Liu *et al*., 2007).

Many people from both industry and research communities have attempted to apply service technologies in e-learning (Zhou *et al*., 2009). However, current research is not comprehensive enough as some of them have only addressed one type of e-learning resource. For example, Simone *et al*. (2005) have proposed a framework to share learning objects using service technologies, and Lucia *et al*. (2008) have developed a model to share learning content together with systems which run using the shared content. However, their work has covered neither the sharing of students' records nor other applications to support assessment tasks.

Some current research about e-learning services is not comprehensive and focuses only on the principle of interoperability. For instance, Phankokkruad and Woraratpanya (2009) have proposed an architecture to allow e-learning services to be better communicated. Sun and Fu (2005) have also investigated the interoperability issues, but have not mentioned other principles such as discoverability or reusability.

3 Educational Services Architecture

Service technologies have been successfully applied in the business domain (Li *et al*., 2009), and we argue that service technologies could be used to share resources in e-learning. The diagram above (Fig. 1) is the design of the educational services architecture we are proposing. Our approach inserts a layer between users and e-learning resource providers – the Educational Services Bus (ESB). There are four main components in the ESB — Service Registry, Message Exchanger, Data Converter and Adapters. — and these are supported by networks and web servers.

3.1 Message Exchanger

The *Message Exchanger* acts as a data transport engine. Its main purpose is to provide message communications for interactions between service clients and providers which carry educational data. This component supports the service-oriented principle of interoperability, and SOAP (Simple Object Access Protocol) is one of the default

communication mechanisms to implement interoperability. Apache Axis2 is a popular technology which supports this (Lucia *et al.,* 2008).

3.2 Data Converter

The main role of a *Data Converter* is to transform the format of educational content in order to reuse it for other purposes. This component supports the principle of reusability in service technologies and tackles problems caused by data format mismatch (Kongdenfha *et al*., 2009). However, currently there are few tools for converting the format of e-learning content (Chung and Chao, 2007). Lack of existing generic conversion tools can be partly explained by the many different data formats for various e-learning content. We suggest that the conversion could contain two stages. The first converts selected content to a transitional format, which is subsequently changed to a specific format which the target service can accept. For example, student essays, which are generated as Word format, are first converted to HTML, and then converted to XML format before another service is able to process them later on for detecting plagiarism (for example).

3.3 Service Registry

The *Service Registry* (also called *Service Broker*), and publishes descriptions of services developed by service providers. This component supports the principle of discoverability. Many technologies have been developed for the Service Registry, and most service deployment platforms, such as the NetBeans IDE or Eclipse supports service publication features (Simone *et al*., 2005).

3.4 Adapters

The purpose of adapters is to support the principle of interoperability. The *Service Adapters* enable platform independence while deploying educational services. They are connectors that map different service interfaces and protocols into a common model which can be accepted by the ESB. A *Client Adapter* (also called a *Client Gateway*) receives requests from clients and routes them to the appropriate components in the ESB. Currently, there is no single approach to the implementation of such adapters, and a variety of technologies have been adopted (Kongdenfha *et al.*, 2009).

3.5 Discussion on ESB

The core component in our educational services bus is the *Message Exchanger*, as it allows other components to be connected together. Adapters are an optional component in our ESB and they are implemented only if the clients or services so require. The integration of these two components allows resources to be better interoperated. Collaboration between the Message Exchanger, Service Registry and the Adapters enable e-learning resources to be better linked and discovered. The cooperation between the Message Exchanger, Data Converter and Adapters enables e-learning resources to be better reused.

Much of the proposed architecture can be implemented by using standard and open source technologies (Sun and Fu, 2005). However, some components such as the Data Converter and the Adapters may require customised implementations because they must cater for a wide variety of different data formats and platforms (Kongdenfha *et al.*, 2009). We suggest that to find optimal, generalisable solutions for implementing these components requires further research.

3.7 Educational Services

Educational services in our research refer to software components that provide certain functionality to support certain learning and teaching related tasks. For each service, its functionalities, operations, data types and binding information are specified in a service interface. In previous research (Yang and Joy, 2009), we conducted a case study within a university and proposed that there should be nine types of educational services. The five services we presented in Figure 1 are for illustrative purposes. Each user can access more than one type of service, and for each type of service there is more than one type of user who can reuse it.

3.6 Use Case Example

The following is an example to explain how our architecture supports the sharing of a typical resource — plagiarism detection software.

Teachers require access to different plagiarism detection applications for different purposes. Sometimes, they need to handle different types of coursework, sometimes they need to use different methods to compare students' assignments against other students' assignments, or against other available web resources. Software for detecting

plagiarism already exists, including the Turnitin (2010) products for essays, and JPlag (2010) and Sherlock (Joy and Luck, 1999) for computer programming assignments. However, users may not know that they exist at all, and they cannot access all of them easily since their user interfaces differ substantially. By using our ESB, plagiarism detection resources can be better described and linked, so that teachers can choose appropriate plagiarism detection software to reuse for different assessment tasks from multiple service providers. The following explains how the various components work together to support the task of detecting plagiarism.

In our architecture, each plagiarism detection tool is wrapped as a service — for example, the Turnitin service, the JPlag service and the Sherlock service. Details about each service are stored and published in the Service Registry. Teachers will first send a request to use a plagiarism detection service to the Message Exchanger, which in turn will contact the Service Registry, and return a number of available services with descriptions. Users can then make selections simply between these services depending on their needs. The Message Exchanger will pass messages containing student assignments to the service that has been selected. The selected service will then process the assignments and return the detection results to users via the Message Exchanger. By using the Data Converter in the ESB, the format of the coursework can then be converted so that it can be easily reused in any plagiarism detection application. As users might access these plagiarism detection services in different locations at different times, and these services might be hosted on different servers by different providers, the current users and available services will change dynamically over time. For these reasons, we also suggest that both Service Adapters and Client Adapters should be adopted in our ESB.

4 CONCLUSIONS AND FUTURE WORK

This paper has proposed the design of our educational services architecture, and has also highlighted potential problems our service approach can solve in current e-learning practices. In the near future, we plan to implement a part of the proposed architecture and conduct an experiment to evaluate if our service approach is actually feasible, in order to better support our research question: how could we share e-learning resources effectively and efficiently using services technologies?

REFERENCES

Bean, J., 2010. *SOA and Web Services Interface Design: Principles, Techniques, and Standards*, Morgan Kaufmann. Amsterdam.

Chung, J., Chao, K., 2007. A View on Service Oriented Architecture, *International Journal on Service-Oriented Computing and Applications*, **1**(2), pp.93-95.

Joy, M.S., Luck, M., 1999. Plagiarism in Programming Assignments, *IEEE Transactions on Education*, **42**(2) pp. 129-133.

JPlag, 2010. https://www.ipd.uni-karlsruhe.de/jplag/, [29th Sep 2010].

Kongdenfha, W., Motahari-Nezhad, H., Benatallah, B., Casati, F., Saint-Paul, R., 2009. Mismatch Patterns and Adaptation Aspects, IEEE Transactions on Services Computing, **2**(2), pp.94-107.

Li, J., Wang, Y., Zhang, J., Zhang, Z., 2009. A Web Service Adapter with Contract Oriented Methodology, *IEEE Conference on Computer and Information Science*, pp.1144-1149.

Liu, J., Wu, Y., Zhao, W., 2007. Modelling Learning Contents Based on Web Services, *International Conference on Next Generation Web Services Practices*, pp.135-140.

Lucia, A., Francese, R., Passero, I., Tortora, G., 2008. A Service Oriented Collaborative Distributed Learning Object Management System. *Lecture Notes in Business Information Processing*, Springer Berlin.

Phankokkruad, M., Woraratpanya, K., 2009. Web services for Learning Management Systems. *International Conference on Communication architecture*, pp.403-408.

Simone, L., Fabio. C., Sean. W., Rubens. N., Saulo. V., 2005. Integrating Repositories of Learning Objects Using Web-Services to Implement Mediators and Wrappers, *International Conference on Next Generation Web Services Practices*, IEEE Computer Society.

Sun, L., Fu, Y., 2005. Interoperability for Elearning Services Management and Provision. *International Journal of World Wide Web*, Springer. Netherlands.

Turnitin, 2010. http://turnitin.com/static/index.html, [29th Sep 2010].

Yang, S., Joy, M., 2008. SOA Services in Higher Education, *International Conference on Education and Educational Technology*, pp. 145-150.

Yang, S., Joy, M., 2009. Designing E-learning Services: A Case Study, *iJAC* **2**(4), pp. 35-43.

Zhou, D., Zhang, Z., Zhong, S., Yao, Y., Zhang, H., 2009, The study of service oriented architecture of E-learning resources and personalized service model. *International Conference on Machine Learning and Cybernetics*, **6**, pp. 3591-3594.